

IMPLEMENTASI SISTEM REKOMENDASI RESEP BERDASARKAN PERSEDIAAN BAHAN DAN PREFERENSI PENGGUNA

Nabilah Argyanti Ardyningrum¹, NamaPenulis², NamaPenulis³

^{1,2} Informatics Engineering, State Polytechnic of Malang

¹nargyanti@gmail.com, ² alamat e-mail, ³ alamat e-mail

Abstract

Memasak adalah aktivitas penting dan menyenangkan yang memungkinkan individu menyajikan hidangan lezat. Namun, menemukan resep yang sempurna yang sejalan dengan bahan-bahan yang tersedia dan preferensi pribadi bisa menjadi tantangan. Untuk mengatasi hal ini, sebuah sistem rekomendasi telah diperkenalkan, yang dikembangkan dengan *content-based filtering* menggunakan *Jaccard similarity*, untuk pemanfaatan bahan-bahan yang dimiliki secara efektif, dan *item-based collaborative filtering* menggunakan *cosine similarity*, untuk mengidentifikasi resep-resep yang sejalan dengan preferensi pengguna berdasarkan preferensi pengguna lain. Pendekatan ini memberikan rekomendasi resep yang akurat dan relevan, memaksimalkan pemanfaatan bahan-bahan dan meningkatkan pengalaman memasak.

Keywords: recommender system, collaborative filtering, content-based filtering, jaccard similarity, cosine similarity

1. Latar Belakang

Memasak adalah aktivitas yang tidak dapat dihindari dalam kehidupan sehari-hari manusia. Setiap harinya, orang-orang memasak untuk memenuhi kebutuhan nutrisi dan menciptakan hidangan yang enak serta bergizi. Selain itu, memasak juga diperlukan untuk mengolah makanan mentah menjadi makanan yang lebih mudah dan aman untuk dikonsumsi. Proses memasak ini membantu menghilangkan bakteri dan patogen potensial dalam makanan, sehingga meningkatkan keselamatan dan kualitas makanan yang dikonsumsi manusia (Doyle & Erickson, 2008).

Saat proses memasak, berbagai bahan makanan diolah menggunakan alat dan teknik memasak yang berbeda, menciptakan beragam rasa dan tampilan hidangan. Alat masak seperti wajan, panci, oven, blender, dan lainnya berperan penting dalam mengolah bahan makanan menjadi hidangan yang lezat dan bergizi. Penggunaan teknik memasak yang tepat, seperti menggoreng, merebus, mengukus, atau memanggang, menghasilkan efek yang berbeda pada bahan makanan. Dalam proses ini, manusia memiliki kesempatan untuk berkreasi dan mengeksplorasi cita rasa, yang pada akhirnya berkontribusi pada tingkat selera makan dan kepuasan mereka terhadap makanan yang telah mereka sajikan (Zhou et al., 2014).

Setiap individu memiliki gaya memasak yang unik, yang sejalan dengan selera makan yang khas. Beberapa orang cenderung menyukai makanan dengan cita rasa tertentu, seperti kesukaan terhadap hidangan pedas, manis, asam, atau gurih (Anzman-Frasca et al., 2018).

Saat seseorang ingin mengonsumsi sesuatu, mereka cenderung mencari makanan yang sesuai dengan ketersediaan bahan makanan, peralatan yang ada, dan preferensi makanan mereka. Selain itu, mereka juga harus mempertimbangkan ketahanan bahan makanan yang dimiliki untuk menghindari pembusukan. Untuk menemukan menu yang sesuai, berbagai metode digunakan, salah satunya adalah mencari resep masakan yang sesuai dengan kriteria-kriteria tersebut. Namun, keterbatasan informasi pada resep-resep yang tersedia sering kali menjadi hambatan, di mana resep tersebut hanya menyajikan informasi terbatas seperti judul, bahan-bahan, dan instruksi (Shirai et al., 2021). Dikarenakan hal tersebut, orang-orang harus melakukan pengecekan satu per satu untuk menentukan kesesuaian dengan situasi dan preferensi mereka. Fenomena ini menyebabkan proses pencarian menu menjadi tidak efisien. Selain itu, kompleksitas dalam memilih makanan yang cocok juga disebabkan oleh banyaknya menu makanan yang memiliki bahan yang serupa, sehingga menyebabkan kebingungan dalam menentukan pilihan.

Dalam menghadapi tantangan tersebut, dibutuhkan pengembangan sistem rekomendasi resep yang efektif dan efisien. Sistem ini menggunakan content-based filtering dengan Jaccard similarity untuk menyajikan rekomendasi berdasarkan ketersediaan bahan masakan, dengan membandingkan bahan-bahan yang dimiliki pengguna dengan bahan-bahan dalam resep. Selain itu, sistem ini juga menerapkan item-based collaborative filtering dengan cosine similarity untuk merekomendasikan resep berdasarkan kesamaan dengan resep yang telah disukai oleh pengguna lain.

Dengan mengadopsi pendekatan content-based filtering dan item-based collaborative filtering, sistem rekomendasi ini diharapkan mampu memberikan rekomendasi resep masakan yang relevan dengan bahan masakan yang dimiliki dan sesuai dengan selera makan pengguna, sehingga proses memasak menjadi lebih efisien dan menyenangkan. Sistem rekomendasi ini juga diharapkan menjadi solusi praktis bagi pengguna yang bingung dalam memilih menu makanan yang sesuai dengan keadaan dan selera, sehingga mereka dapat dengan mudah menemukan inspirasi untuk mencoba masakan baru yang cocok dengan preferensi pribadi mereka.

2. Related Works

Bagian ini membahas beberapa studi dan karya dalam literatur yang relevan dengan topik ini.

Shilpa Chaudhari, Aparna R., Vinay G Tekkur, Pavan G L., dan Shreekanth R Karki mengembangkan Ingredient/Recipe Algorithm using Web Mining and Web Scraping for Smart Chef, sebuah sistem rekomendasi resep menggunakan web scraping yang dapat membantu pengguna mencari rekomendasi resep berdasarkan bahan yang dimasukkan dalam kata kunci pencarian (Chaudhari et al., 2020).

Ifeoma Adaji, Czarina Sharmaine, Simone Debrowney, Kiemute Oyibo, dan Julita Vassileva mengembangkan Personality Based Recipe Recommendation Using Recipe Network Graphs. Sistem rekomendasi resep menggunakan tipe kepribadian dari pemberi ulasan resep dapat membantu pengguna lain mendapatkan rekomendasi resep. Pengguna dengan tipe kepribadian yang mirip cenderung menyukai resep yang sama (Adaji et al., 2018).

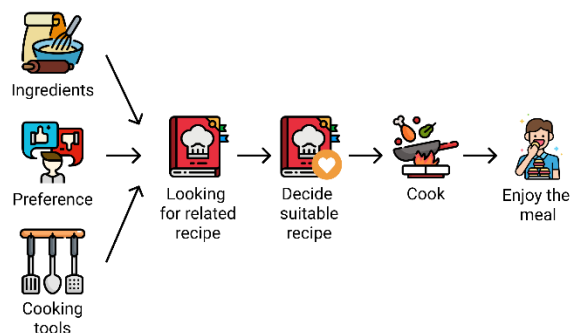
Nilesh, Madhu Kumari, Pritom Hazarika, dan Vishal Raman mengembangkan Recommendation of Indian Cuisine Recipes Based on Ingredients, sebuah sistem rekomendasi resep menggunakan web scraping dan metode berbasis konten yang dapat memberikan rekomendasi untuk hidangan India berdasarkan bahan pada resep (Nilesh et al., 2019).

3. Tinjauan Pustaka

Dalam bagian ini, beberapa teori akan dibahas untuk memberikan pemahaman yang lebih mendalam tentang topik ini.

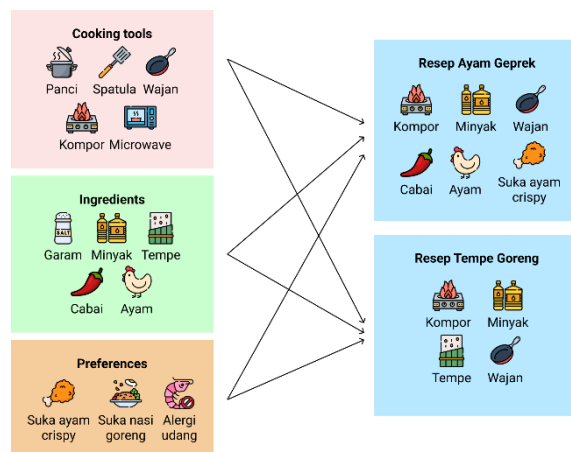
3.1 Memasak dengan Resep

Memasak dari sebuah resep melibatkan langkah-langkah untuk menciptakan hidangan yang lezat dan memuaskan. Memasak dengan resep dapat dimulai dengan mengumpulkan berbagai elemen, seperti bahan, peralatan, dan preferensi. Selanjutnya, resep dapat dicari sesuai dengan bahan, peralatan, dan preferensi. Setelah resep dipilih, resep bisa diterapkan dengan menggunakan bahan-bahan dan peralatan yang tercantum pada resep. Setelah memasak selesai, makanan siap untuk dinikmati (Kadowaki et al., 2014).



Gambar 1. Langkah-langkah memasak dengan resep

Proses mencari resep memerlukan banyak waktu. Adanya resep yang mirip mengharuskan individu mempertimbangkan berbagai faktor untuk menemukan yang cocok. Beberapa individu cenderung mempertimbangkan bahan-bahan yang tersedia sebelum mencari resep, sementara yang lain lebih mengutamakan preferensi pribadi mereka. Hal ini menyoroti kompleksitas dalam mencari resep yang sesuai dengan kebutuhan dan selera masing-masing individu (Tobey et al., 2019).



Gambar 2. Pertimbangan kecocokan resep

3.2 Struktur Resep

Struktur sebuah resep sangat penting dalam dunia kuliner, karena menjadi panduan bagi para pembuat makanan selama proses memasak. Secara umum, sebuah resep terdiri dari beberapa komponen utama. Pertama, ada judul yang menggambarkan hidangan yang akan disiapkan. Selanjutnya, bagian bahan menyediakan daftar komponen yang diperlukan dengan ukuran yang tepat. Instruksi membentuk inti dari resep, memberikan panduan langkah demi langkah mulai dari persiapan hingga penyajian, termasuk teknik memasak dan saran penyajian. Memahami struktur resep memungkinkan individu untuk menciptakan hidangan yang lezat dan seimbang (Kadowaki et al., 2014).

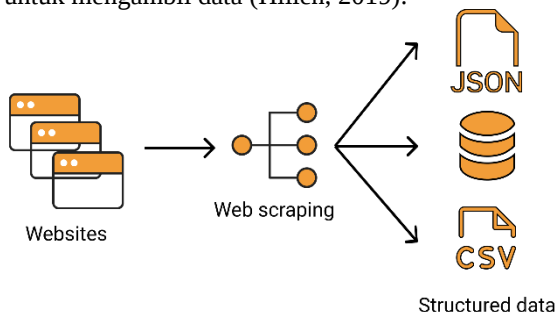
3.3 Pengumpulan Data

A. Web Scraping

Web scraping adalah teknik pengambilan informasi dari halaman web secara otomatis yang memungkinkan pengumpulan data dari berbagai sumber, di mana data tersebut diekstrak ke dalam file atau database untuk akses dan analisis lebih lanjut (Lawson, 2015).

Saat menjalankan web scraping, sebuah request akan dikirimkan ke server dan server akan mengirimkan response berupa data yang diperlukan. Data pada response tersebut kemudian di-parse dan diekstrak sesuai dengan data yang dibutuhkan. Teknik parsing ini memungkinkan sistem untuk mengenali dan mengambil bagian-bagian penting dari halaman web, seperti teks, tabel, gambar, dan lainnya (Zhao, 2017).

Web scraping umumnya digunakan untuk analisis statistik, sistem prediksi, dan *data training* pada *machine learning*, terutama dalam kasus di mana situs web hanya menampilkan informasi yang tidak dapat diunduh secara langsung. Hal ini membuat web scraping menjadi metode yang efisien untuk mengambil data (Hillen, 2019).



Gambar 3. Proses web scraping

B. Data Filtering dan Tokenization pada Text Preprocessing

Data filtering dan *tokenization* adalah langkah penting dalam *text preprocessing*. Langkah-

langkah ini membantu mengubah data mentah menjadi format yang lebih terstruktur dan mudah dikelola untuk analisis (Bhujade & Janwe, 2011).

Data filtering berfokus pada penghapusan data yang tidak relevan, termasuk menghilangkan karakter khusus, tanda baca, atau nilai yang hilang. Langkah ini memastikan bahwa data yang digunakan untuk pemodelan atau analisis akurat dan konsisten (Manning et al., 2008).

Tokenization melibatkan pemecahan teks menjadi unit-unit yang lebih kecil, seperti kata-kata atau frasa, yang disebut sebagai *token*. Proses ini sangat berguna dalam analisis teks, *natural language processing*, dan tugas-tugas berbasis teks lainnya. Dengan mengalikan teks, menjadi lebih mudah untuk mengekstrak informasi yang bermakna dan melakukan berbagai operasi terkait teks (Manning et al., 2008).

3.4 Metode Pencocokan Melalui Rekomendasi Resep

A. Content-based Filtering dengan Jaccard Similarity

Content-based filtering merupakan metode rekomendasi yang memanfaatkan konten dari item yang sudah diketahui untuk merekomendasikan item-item serupa. Metode ini melibatkan informasi seperti deskripsi, tag, atau atribut lain yang relevan dengan item-item tersebut (Sukestiyarno et al., 2023).

Jaccard similarity adalah salah satu metode yang digunakan dalam *content-based filtering*. Metode ini mengukur kesamaan antara dua item dengan menghitung sejauh mana elemen-elemen tersebut serupa atau tumpang tindih.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Dengan menggunakan *Jaccard similarity*, item-item yang memiliki tingkat kemiripan yang tinggi dapat direkomendasikan kepada pengguna (Bhatia, 2018).

B. Item-based Collaborative Filtering dengan Cosine Similarity

Item-based Collaborative Filtering adalah metode dalam sistem rekomendasi yang berfokus pada kesamaan antara item dalam kumpulan data. Metode ini mencari item yang mirip dengan item yang disukai oleh pengguna. Dengan menggunakan *Item-based Collaborative Filtering*, sistem dapat memberikan rekomendasi item berdasarkan preferensi pengguna lain yang memiliki selera yang mirip. Metode ini berguna dalam merekomendasikan item seperti produk, film, buku, atau resep kepada pengguna (Jannach, 2011).

Dalam sistem rekomendasi, metode ini mengukur tingkat kesamaan antara item-item dalam

kumpulan data. Semakin tinggi nilai *cosine similarity* antara dua item, semakin mirip keduanya.

$$\cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

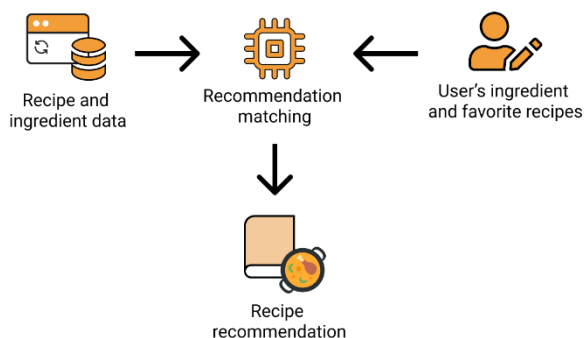
Dengan menggunakan metode ini, sistem rekomendasi dapat menemukan item yang memiliki kemiripan yang tinggi dengan preferensi pengguna, sehingga memberikan rekomendasi yang relevan. Pengguna dapat menemukan item-item baru dan menarik berdasarkan kesamaannya dengan item yang disukai oleh pengguna lain yang memiliki preferensi serupa (Dangeti, 2017).

4. Desain Sistem Rekomendasi Resep

Bagian ini berfokus pada proses desain dan pengembangan sistem rekomendasi resep.

4.1 Gambaran Umum

Proses rekomendasi resep melibatkan data resep dan bahan makanan, ketersediaan bahan makanan, serta resep yang disukai dari masukan pengguna. Data-data tersebut dijadikan acuan untuk algoritma rekomendasi guna menghasilkan saran resep bagi pengguna. Dengan menerapkan pendekatan ini, sistem dapat menawarkan rekomendasi resep yang sesuai dengan preferensi dan kebutuhan pengguna.



Gambar 4. Gambaran alur logika sistem rekomendasi resep

4.2 Pengumpulan Data Resep

A. Web Scraping

Pada tahap ini, dilakukan ekstraksi informasi yang spesifik mengenai setiap resep. Informasi ini digunakan untuk menyajikan detail lengkap mengenai setiap resep, di mana informasi ini didapat dari website masakapahariini. Data-data yang diekstrak antara lain:

1. Title: judul resep.
2. Source URL: sumber tempat resep diperoleh.
3. Image URL: tautan berupa gambar hidangan.

4. Servings: jumlah porsi yang dihasilkan dari resep.
5. Time: waktu yang dibutuhkan dalam memasak.
6. Difficulty: tingkat kesulitan dalam memasak.
7. Ingredients: daftar bahan yang diperlukan dalam memasak.
8. Instructions: langkah-langkah yang harus diikuti dalam memasak.

Untuk melakukan web scraping, langkah-langkah yang harus diikuti adalah sebagai berikut. Pertama, halaman detail resep diakses melalui URL detail resep. Konten halaman kemudian di-*parsing* untuk mengidentifikasi elemen-elemen penting yang ingin diambil. Selanjutnya, variabel didefinisikan untuk menyimpan data yang akan diambil dari halaman tersebut. Misalnya, variabel "title" dibuat untuk menyimpan judul resep, "ingredients" untuk menyimpan daftar bahan-bahan, dan "instructions" untuk menyimpan langkah-langkah instruksi memasak. Terakhir, data yang berhasil diekstrak akan disimpan dalam format yang sesuai dengan kebutuhan.

Contoh data resep yang masih berbentuk HTML dapat dilihat pada Tabel 1. Hasil data resep yang telah berhasil diambil menggunakan *web scraping* ditampilkan dalam Tabel 2. Dari proses scraping ini, beberapa komponen telah didefinisikan, seperti title, source URL, image URL, servings, time, difficulty, ingredients, dan instructions yang berhasil diambil.

Tabel 1. Contoh data bahan pada website masakapahariini dalam bentuk HTML

<div class="d-flex"> <div >7<="" <div="" class="item" data-base-measurement-unit="" data-base-quantity="7" data-value="1.17" div>=""> siung bawang putih </div> </div>	
<div class="d-flex"> <div >11<="" <div="" class="item" data-base-measurement-unit="" data-base-quantity="11" data-value="1.17" div>=""> butir bawang merah </div> </div>	

Tabel 2. Contoh data setelah dilakukan web scraping

title	source_url	image_url	time	servings	difficulty	ingredients	Instructions
Resep Renda Ayam	https://www.masakapahariini.com/m/resep/resep-ayam-rendan/	https://www.masakapahariini.com/wp-content/uploads/2018/10/ayam-rendan-g-MAHI-4-500x300.jpg	2jam	6	Cukup	[2 sdm minyak sayur', '7 siung bawang putih', '11 butir bawang merah', '3 cm jahe', '5 buah cabai merah']	[Tumis bumbu halus hingga harum. Tambahkan air dan santan. Masukkan potongan ayam. Masak hingga hingga air menyusut setengah.', 'Masukkan kacang merah. Masak kembali dengan api kecil hingga mengental.', 'Angkat dan hidangkan di piring saji.']

B. Text Filtering untuk Data Resep

Text filtering untuk data resep bertujuan untuk membuat data lebih mudah dipahami, terstruktur, dan ringkas. Proses ini dilakukan dengan memahami pola kalimat di setiap kolom data, yang dilanjutkan dengan membersihkan teks menggunakan ekspresi reguler (regex), dan mengekstrak teks yang relevan. Hasilnya yaitu berupa data resep yang lebih mudah untuk dikelola lebih lanjut.

Tabel 3. Contoh data resep sebelum dan sesudah dilakukan text filtering

Before

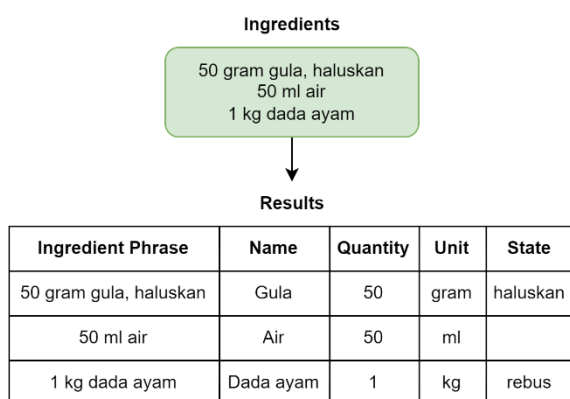
title	calories	time
Resep Ayam Sambal Geprek, Pedasnya Menggugah Selera	400Kkal	30mnt
Cara Membuat Ketupat, Sajian Wajib Saat Lebaran	700Kkal	1jam 30 mnt

After

title	calories	time
Ayam Sambal Geprek	400	30
Ketupat	700	90

C. Tokenization untuk Data Bahan

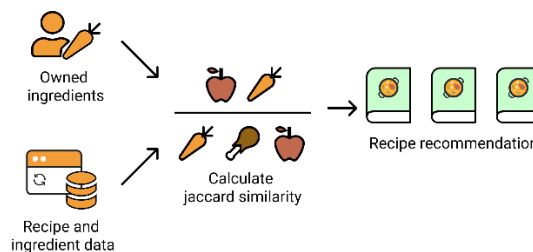
Sebelum data bahan digunakan sebagai data awal untuk menentukan rekomendasi makanan, langkah awal yang dilakukan adalah memisahkan data bahan menjadi empat bagian utama, yaitu nama bahan, kuantitas, satuan, dan state, menggunakan metode *tokenization*. Proses pemisahan ini dilakukan dengan menggunakan *regular expression* (regex) sehingga data dapat diorganisir dengan struktur yang sesuai dan sesuai dengan pola yang diinginkan.



Gambar 5. Contoh data sebelum dan sesudah dilakukan *tokenization*

4.3 Proses Rekomendasi Resep

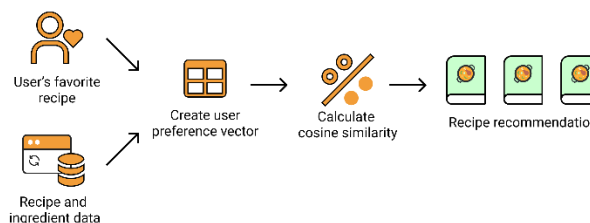
A. Rekomendasi berdasarkan Ketersediaan Bahan



Gambar 6. Proses *content-based filtering* untuk rekomendasi resep berdasarkan ketersediaan bahan pengguna

Langkah pertama dalam proses rekomendasi resep menggunakan *content-based filtering* adalah mengidentifikasi bahan-bahan yang dimiliki pengguna. Selanjutnya, sistem membandingkan kumpulan bahan-bahan yang dimiliki oleh pengguna dengan kumpulan bahan-bahan dari setiap resep menggunakan *Jaccard similarity*. *Jaccard similarity* mengukur tingkat kemiripan bahan pengguna dengan bahan resep. Setelah perbandingan dilakukan, sistem mengidentifikasi dan merekomendasikan resep yang disortir berdasarkan nilai *Jaccard similarity* tertinggi, di mana resep dengan peringkat teratas merupakan resep dengan bahan yang paling banyak dimiliki oleh pengguna.

B. Rekomendasi Berdasarkan Resep yang Disukai Pengguna



Gambar 7. Proses *collaborative filtering* untuk rekomendasi resep berdasarkan resep yang disukai pengguna

Pada metode *collaborative filtering*, diperlukan masukan dari pengguna berupa resep yang disukai oleh semua pengguna, serta data resep beserta bahan. Data-data ini digunakan untuk membangun profil preferensi pengguna dalam bentuk vektor.

Selanjutnya, ketika pengguna ingin mendapatkan rekomendasi, sistem membandingkan resep yang disukai oleh pengguna yang bersangkutan dengan resep yang disukai oleh pengguna lain dengan *cosine similarity*. Semakin besar nilai *similarity*-nya, maka nilai rekomendasi akan semakin tinggi.

Setelah perbandingan dilakukan, sistem merekomendasikan resep-resep yang memiliki kemungkinan cukup besar sesuai dengan selera pengguna yang sedang mengakses sistem tersebut.

5. Implementasi Sistem

Bagian ini menjelaskan implementasi dari sistem yang diusulkan, termasuk pengembangan sistem dan hasil rekomendasi.

5.1 Arsitektur Website

Sistem rekomendasi resep ini dikembangkan dengan menggunakan Django Rest Framework sebagai *backend* untuk berkomunikasi dengan *client* dan *database*. Pada sisi *frontend*, Vue.js digunakan untuk menyajikan antarmuka pengguna yang responsif dan interaktif.



Gambar 8. Website Architecture

Pertukaran data melalui permintaan HTTP menggunakan *library* Axios. Django Rest Framework memproses permintaan dari Vue.js, kemudian mengelola data yang diperlukan. *Response* yang dihasilkan dari pengelolaan data tersebut kemudian dikirim kembali ke Vue.js untuk menampilkan data yang telah diproses kepada pengguna.

5.2 Implementasi Web Scraping

Proses *web scraping* dilakukan dengan bahasa pemrograman Python melalui module *recipe_scrapers*. Module ini dapat mengekstraksi yang informasi yang diperlukan, yaitu dengan cara membuat file .csv dengan header detail resep yang dilanjutkan dengan ekstrasi data tiap detail resep.

Tabel 4. Potongan kode web scraping

```

from recipe_scrapers import scrape_me
import random
import csv

def csv_store(linkList, csvfile):
    header = ["id", "title", "time", "servings", "ingredients",
              "instructions", "image_url", "source_url"]
    with open(csvfile, 'w', encoding='UTF8') as f:
        writer = csv.writer(f)
        writer.writerow(header)

    for link in linkList:
        scraper = scrape_me(link, wild_mode=True)
        id = random.randint(1000000, 9999999)
        row = [id, scraper.title(), scraper.total_time(),
              scraper.yields(), scraper.ingredients(),
              scraper.instructions(), scraper.image(),
              scraper.links()]
        writer.writerow(row)

links = [
    "https://www.masakapahariini.com/resep/resep-lontong-kari-
    sapi/",
    # Other URLs
]

csv_store(links, "resep1.csv")
  
```

5.3 Text Preprocessing untuk Data Resep dan Data Bahan

Text preprocessing dilakukan pada Python menggunakan module *regular expression* (regex). Hasil pembersihan data pada resep melalui *text filtering* dan pemisahan data bahan melalui *tokenization* disimpan ke dalam file .csv dan diimpor ke database MySQL.

5.4 Recipe Recommendation

Algoritma *content-based filtering* didapatkan dengan menghitung *Jaccard similarity* tiap resep, dengan membandingkan bahan yang dimiliki pengguna dengan bahan pada resep. Algoritma ini ditulis dengan bahasa pemrograman Python yang diletakkan pada *framework* Django.

Tabel 5. Cuplikan kode *content-based filtering*

```
def jaccard_similarity(user_ingredients, recipe_ingredients):
    intersection =
    len(user_ingredients.intersection(recipe_ingredients))
    union = len(user_ingredients.union(recipe_ingredients))
    similarity_score = intersection / union
    return similarity_score

def get_recommendation_by_ingredients(user):
    pantry =
    set(Pantry.objects.filter(user=user).values_list('raw_ingredient_id', flat=True))
    # Get valid recipes
    recipe_ingredients = Ingredient.objects.filter(
        raw_ingredient__pantry__user=user
    ).values('recipe_id', 'raw_ingredient_id', 'id',
    user_id=F('raw_ingredient__pantry__user_id'))

    data = {}
    for ingredient in recipe_ingredients:
        user_id = ingredient['user_id']
        recipe_id = ingredient['recipe_id']
        raw_ingredient_id = ingredient['raw_ingredient_id']

        if recipe_id in data:
            data[recipe_id]['raw_ingredient_ids'].add(raw_ingredient_id)
        else:
            data[recipe_id] = {
                'user_id': user_id,
                'recipe_id': recipe_id,
                'raw_ingredient_ids': {raw_ingredient_id}
            }

    for recipe_id, recipe_data in data.items():
        raw_ingredient_ids =
        set(Ingredient.objects.filter(recipe_id=recipe_id).values_list('raw_ingredient_id', flat=True))
        owned_raw_ingredient =
        set(get_owned_recipe_ingredients(user, recipe_id))
        similarity = jaccard_similarity(
            owned_raw_ingredient, pantry, raw_ingredient_ids)

        recipe_data['similarity'] = similarity * 100

    sorted_recipes = sorted(
        data.values(), key=lambda x: x['similarity'],
        reverse=True)

    return sorted_recipes
```

Algoritma *item-based collaborative filtering* didapatkan dengan menghitung *cosine similarity* tiap resep, yang membandingkan resep favorit pengguna yang sedang masuk ke dalam sistem dengan resep yang disukai oleh semua pengguna. Algoritma ini juga ditulis dengan bahasa pemrograman Python yang diletakkan pada *framework* Django.

Tabel 6. Cuplikan kode *item-based collaborative filtering*

```
def calculate_cosine_similarity():
    interaction_matrix = create_interaction_matrix()
    similarity_matrix = cosine_similarity(interaction_matrix.T)
    similarity_df = pd.DataFrame(
        similarity_matrix, index=interaction_matrix.columns,
        columns=interaction_matrix.columns)
    return similarity_df, interaction_matrix

def get_recommendations_for_user(user_id):
    similarity_matrix, interaction_matrix =
    calculate_cosine_similarity()

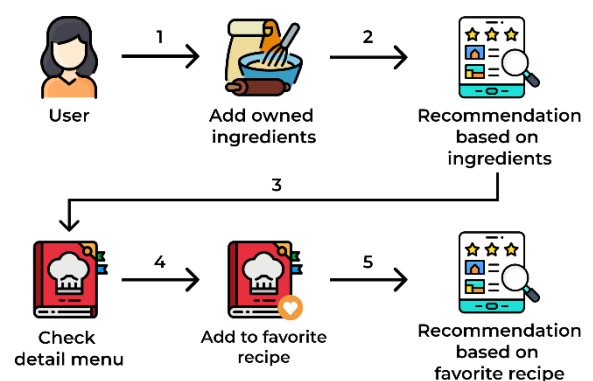
    if user_id in interaction_matrix.index:
        scores = np.dot(interaction_matrix.values,
            similarity_matrix)

        recommended_items = [interaction_matrix.columns[i]
            for i in
            np.argsort(scores[interaction_matrix.index == user_id,
                :])[0][::-1]]

        recommendations = []

        for item in recommended_items:
            score = scores[interaction_matrix.index ==
                user_id, interaction_matrix.columns
                == item][0]
            similarity_score = scores[interaction_matrix.index
                == user_id,
                interaction_matrix.columns == item][0]
            if score > 0:
                recommendation = {
                    'recipe_id': item,
                    'similarity': round(similarity_score, 2),
                    'score': round(score, 2)
                }
                recommendations.append(recommendation)
        return recommendations
    else:
        return []
```

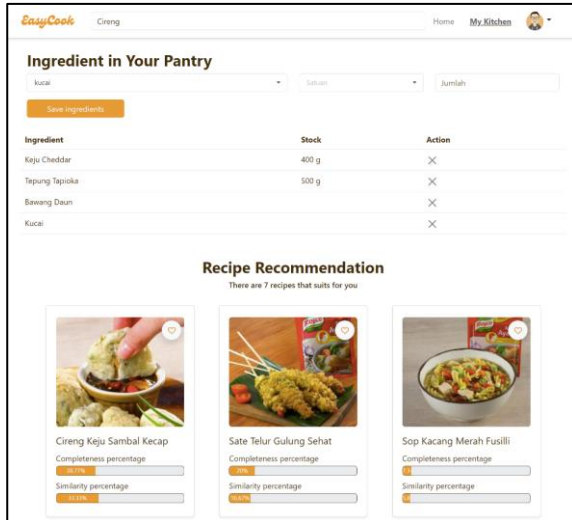
Hasil dari kedua algoritma ini disajikan dalam bentuk daftar resep yang telah disortir berdasarkan rekomendasi tertinggi, yang ditampilkan kepada pengguna melalui *website*.



Gambar 9. Alur penggunaan sistem

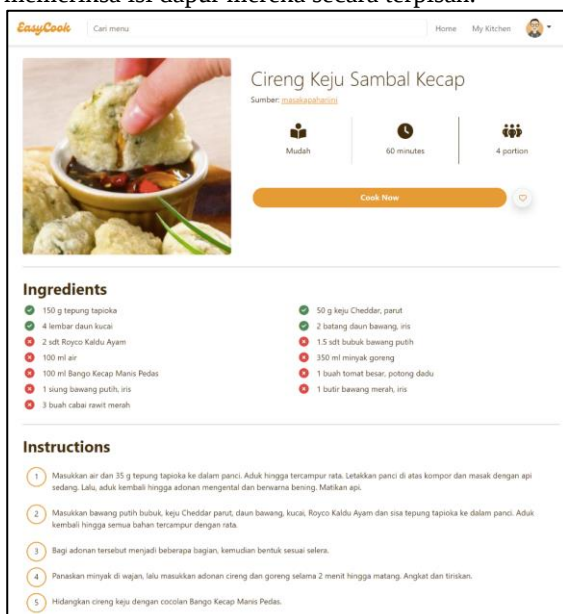
Untuk menggunakan sistem ini, pengguna diminta untuk memasukkan bahan-bahan yang mereka miliki ke dalam *website*. Setelah sistem

mengetahui daftar bahan yang dimiliki pengguna, sistem akan menyarankan resep-resep yang sesuai dengan bahan-bahan yang tersedia. Hal ini memudahkan pengguna untuk menemukan resep-resep yang dapat mereka masak dengan bahan yang mereka miliki.



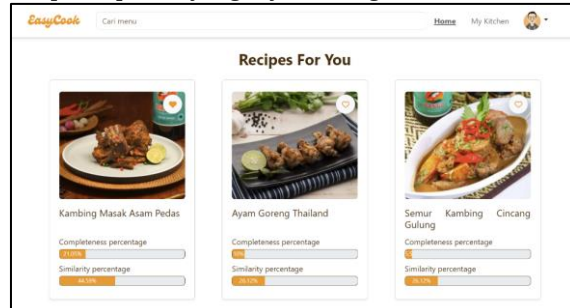
Gambar 10. Halaman My Pantry

Informasi mengenai bahan-bahan yang dimiliki oleh pengguna juga ditampilkan dalam detail resep, di mana bahan-bahan yang dimiliki ditandai dengan tanda centang, dan yang belum dimiliki dengan tanda silang. Dengan cara ini, pengguna dapat dengan mudah mengidentifikasi bahan-bahan yang sudah mereka miliki dan yang perlu mereka peroleh untuk menyiapkan resep tertentu. Fitur ini meningkatkan pengalaman pengguna dan mempercepat proses memasak, karena pengguna dapat dengan cepat menentukan apakah mereka dapat memasak resep tanpa harus memeriksa isi dapur mereka secara terpisah.



Gambar 11. Halaman detail resep

Selain itu, jika pengguna menyukai resep tertentu, halaman utama akan memberikan rekomendasi untuk resep-resep yang mirip dengan resep yang disukainya. Fitur rekomendasi personal ini bertujuan untuk memperkenalkan pengguna pada resep-resep baru yang sejalan dengan selera mereka.



Gambar 12. Halaman Home

6. Evaluasi

Bagian ini menjelaskan evaluasi terhadap pengembangan sistem yang telah dilakukan.

6.1 Evaluasi Web Scraping

Pada bagian ini, pengimplementasian web scraping dilakukan menggunakan perpustakaan `recipe_scraper` dalam bahasa Python untuk mengumpulkan data resep dari masakapahariini. Setelah data terkumpul, informasi resep dibersihkan untuk menghilangkan rincian yang tidak perlu. Data bahan juga diproses, dengan memisahkan nama-nama bahan, satuan, kuantitas, dan statusnya.

Implementasi *web scraping* ini menghasilkan 1591 resep, dengan 22606 bahan yang terkumpul dari semua resep. Setelah dikelompokkan, sebanyak 913 bahan unik dengan nama dan karakteristik yang berbeda berhasil diidentifikasi.

6.2 Evaluasi Rekomendasi Resep Berdasarkan Ketersediaan Bahan

Sistem ini diujikan kepada 47 individu, di mana kemudian diambil 4 pengguna terpilih sebagai evaluasi. Pengguna-pengguna ini diminta untuk memasukkan bahan-bahan resep mereka ke dalam sistem. Tujuan utama dari pengujian ini adalah menganalisis rekomendasi resep yang dihasilkan dan mengidentifikasi faktor-faktor yang memengaruhi hasil rekomendasi.

Selama evaluasi, performa sistem dalam memberikan rekomendasi resep berdasarkan bahan-bahan yang dimasukkan oleh setiap pengguna diamati. Dua aspek penting yang khusus diperhatikan adalah kelengkapan bahan dan jumlah rekomendasi resep. Kelengkapan bahan mengacu pada sejauh mana bahan-bahan yang dimasukkan oleh pengguna cocok dengan bahan-bahan resep.

Tabel 7. Hasil evaluasi rekomendasi resep berdasarkan ketersediaan bahan pengguna

	User 1	User 2	User 3	User 4
Jumlah bahan yang dimasukkan	16	8	6	6
Jumlah rekomendasi yang muncul	969	1149	533	1171
Persentase kelengkapan bahan tertinggi	46.15	38.89	28.57	42.86
Bahan yang dimiliki pengguna pada resep tertinggi	Udang, daun bawang, wortel, minyak wijen, telur, jamur shiitake	Kentang, garam, ayam, mentega, bawang putih	Ayam, cabai	Ayam, telur, nasi
Jumlah bahan pada resep tertinggi	13	16	7	7
Jumlah bahan yang dimiliki pada resep tertinggi	6	5	2	3

Berdasarkan data yang diberikan, terdapat beberapa temuan menarik dalam analisis rekomendasi resep untuk pengguna-pengguna tertentu.

Pertama, User 1 memiliki kelengkapan bahan tertinggi, mencapai 46%, diikuti oleh User 4 dengan 42,86%. Persentase kelengkapan bahan dipengaruhi oleh sejauh mana bahan-bahan pengguna cocok dengan bahan-bahan dalam resep. Sebagai contoh, User 1 memiliki 6 bahan yang cocok dari total 13 bahan dalam resep, sehingga menghasilkan persentase kelengkapan sebesar 46,15%. Sementara itu, User 4 memiliki 3 bahan yang cocok dari total 7 bahan dalam resep, menghasilkan persentase kelengkapan sebesar 42,86%. Meskipun perbedaannya sedikit, keduanya memiliki bahan-bahan yang cocok dengan jumlah bahan dalam resep.

Kedua, dapat diamati bahwa jumlah bahan yang dimasukkan tidak selalu mempengaruhi jumlah

rekomendasi resep. Sebagai contoh, baik User 3 maupun User 4 memasukkan 6 bahan, tetapi jumlah rekomendasi resep untuk keduanya berbeda signifikan, yaitu 533 dan 1171 resep. Faktor lain seperti bobot kemiripan dan kesesuaian bahan dengan resep juga memainkan peran dalam menentukan hasil rekomendasi.

Ketiga, menarik untuk dicatat bahwa User 2 dan User 1 hampir dua kali lipat jumlah bahan yang dimasukkan, namun User 2 menerima lebih banyak rekomendasi resep. User 2 memasukkan 8 bahan dan mendapatkan 1149 rekomendasi resep, sementara User 1, dengan 16 bahan, hanya menerima 969 rekomendasi resep. Hal ini menunjukkan bahwa jumlah bahan yang dimasukkan oleh pengguna tidak selalu menjadi faktor utama dalam menentukan jumlah rekomendasi resep.

6.3 Evaluasi Rekomendasi Resep Berdasarkan Resep yang Disukai Pengguna

Sistem ini diuji melibatkan empat pengguna sebagai pengujinya, terdiri dari dua ibu rumah tangga yang aktif memasak untuk keluarga mereka, dan dua individu yang tinggal sendiri. Setiap pengguna diminta untuk menyukai beberapa resep sebagai bagian dari proses pengujian. Setelah itu, sistem memberikan rekomendasi berdasarkan selera pengguna, dan pengguna diminta untuk memilih rekomendasi yang sesuai dengan selera mereka. Hasil evaluasi ini akan dianalisis dan dibandingkan untuk mengevaluasi seberapa baik sistem dapat memberikan rekomendasi resep yang akurat dan personal.

	User 1	User 2	User 3	User 4
Status pengguna	Married	Married	Single	Single
Jumlah resep favorit pengguna	10	6	3	3
Jumlah rekomendasi yang muncul	8	11	11	8
Jumlah rekomendasi yang relevan	6	7	9	5

Berdasarkan data pengujian yang diberikan, terdapat empat pengguna yang terlibat, masing-masing dengan status perkawinan yang berbeda, yaitu dua pengguna yang sudah menikah (User 1 dan User 2) dan dua pengguna yang lajang (User 3 dan User 4).

Pertama, dari segi jumlah rekomendasi yang diberikan oleh sistem, terdapat variasi yang signifikan. User 2 dan User 3 menerima jumlah

rekomendasi tertinggi, masing-masing mendapatkan 11 resep. Di sisi lain, User 1 dan User 4 menerima jumlah rekomendasi yang sedikit lebih rendah, masing-masing mendapatkan 8 resep. Hal ini mengindikasikan bahwa sistem memberikan jumlah rekomendasi yang berbeda kepada setiap pengguna, kemungkinan berdasarkan preferensi dan riwayat penilaian resep mereka.

Kedua, jumlah rekomendasi yang relevan berdasarkan preferensi pengguna juga bervariasi. User 3 memiliki jumlah rekomendasi yang relevan tertinggi, yaitu 9 resep, sedangkan User 4 memiliki jumlah terendah, hanya 5 resep. Ini menunjukkan bahwa sistem dapat memberikan rekomendasi yang lebih cocok untuk Pengguna 3 dibandingkan dengan Pengguna 4.

Ketiga, terdapat perbedaan yang mencolok dalam jumlah resep favorit yang disukai oleh masing-masing pengguna. User 1 memiliki resep favorit terbanyak, yaitu 10 resep, sementara User 3 dan User 4 hanya menyukai 3 resep. User 2 berada di tengah-tengah dengan 6 resep favorit. Hal ini menunjukkan variasi dalam preferensi kuliner dan minat di antara para pengguna.

6.4 Analisis Evaluasi

Hasil evaluasi mengungkapkan beberapa temuan menarik mengenai rekomendasi resep berdasarkan bahan-bahan. Meskipun jumlah bahan yang dimasukkan oleh pengguna memainkan peran penting, faktor lain seperti kesesuaian bahan dengan resep juga mempengaruhi jumlah rekomendasi resep. Selain itu, jenis bahan yang disediakan, baik bahan dasar seperti air, garam, dan bawang merah, atau bahan-bahan khusus seperti wortel, cabai merah, dan jamur shiitake, juga memengaruhi jumlah rekomendasi resep yang diberikan kepada pengguna.

Selanjutnya, dalam evaluasi rekomendasi resep berdasarkan preferensi pengguna, ditemukan bahwa setiap pengguna memiliki preferensi yang berbeda untuk resep-resep yang mereka sukai. Beberapa pengguna mungkin menyukai banyak resep, sementara yang lain hanya lebih suka beberapa resep saja. Menariknya, jumlah resep favorit yang disukai oleh pengguna tidak mempengaruhi jumlah rekomendasi yang muncul. Sebagai contoh, User 1 menyukai 10 resep tetapi hanya menerima 8 rekomendasi, sementara User 3, yang hanya menyukai 3 resep, mendapatkan 11 rekomendasi. Selain itu, variasi jumlah rekomendasi yang relevan mengindikasikan keragaman preferensi pengguna. Beberapa pengguna memiliki preferensi yang hampir identik dengan rekomendasi yang diberikan, sementara yang lain memiliki preferensi yang sedikit berbeda. Secara keseluruhan, hasil evaluasi menunjukkan bahwa sistem rekomendasi resep ini cukup akurat, dengan lebih dari 50% dari

rekomendasi yang diberikan sesuai dengan preferensi pengguna.

7. Kesimpulan

Sistem rekomendasi resep berdasarkan bahan-bahan dan preferensi pengguna terbukti relevan dan efektif. Dalam mengevaluasi rekomendasi resep berdasarkan bahan-bahan, ditemukan bahwa jumlah dan jenis bahan yang dimasukkan oleh pengguna memengaruhi jumlah resep yang direkomendasikan, dan kesesuaian bahan dengan resep juga memengaruhi rekomendasi yang muncul. Di sisi lain, meskipun setiap pengguna memiliki preferensi yang berbeda untuk resep yang disukai, hasil evaluasi rekomendasi resep berdasarkan preferensi pengguna menunjukkan bahwa rekomendasi yang diberikan tetap relevan dengan preferensi pengguna. Hal ini mengindikasikan bahwa sistem rekomendasi berhasil menyajikan resep-resep yang sesuai dengan keinginan dan selera pengguna, menjadikannya sebagai alat bantu memasak yang sangat berguna.

Referensi:

- Adaji, I., Sharmaine, C., Debrowney, S., Oyibo, K., & Vassileva, J. (2018). Personality Based Recipe Recommendation Using Recipe Network Graphs. In G. Meiselwitz (Ed.), *Social Computing and Social Media Technologies and Analytics* (Vol. 10914, pp. 161–170). Springer International Publishing. https://doi.org/10.1007/978-3-319-91485-5_12
- Anzman-Frasca, S., Ventura, A. K., Ehrenberg, S., & Myers, K. P. (2018). Promoting healthy food preferences from the start: A narrative review of food preference learning from the prenatal period through early childhood: Promoting healthy food preferences. *Obesity Reviews*, 19(4), 576–604. <https://doi.org/10.1111/obr.12658>
- Bhatia, A. (2018). *Machine Learning in Java: Helpful Techniques to Design, Build, and Deploy Powerful Machine Learning Applications in Java, 2nd Edition* (2nd revised ed). Packt Publishing Ltd.
- Bhujade, V., & Janwe, N. J. (2011). Knowledge Discovery in Text Mining Technique Using Association Rules Extraction. 2011 *International Conference on Computational Intelligence and Communication Networks*, 498–502. <https://doi.org/10.1109/CICN.2011.104>
- Chaudhari, S., Aparna, R., Tekkur, V. G., Pavan, G. L., & Karki, S. R. (2020). Ingredient/Recipe Algorithm using Web Mining and Web Scraping for Smart Chef. 2020 *IEEE International Conference on*

- Electronics, Computing and Communication Technologies (CONECCT)*, 1–4.
<https://doi.org/10.1109/CONECCT50063.2020.9198450>
- Dangeti, P. (2017). *Statistics for machine learning: Techniques for exploring supervised, unsupervised, and reinforcement learning models with Python and R*. Packt Publishing.
- Doyle, M. P., & Erickson, M. C. (2008). Summer meeting 2007 the problems with fresh produce: An overview. *Journal of Applied Microbiology*, 105(2), 317–330.
<https://doi.org/10.1111/j.1365-2672.2008.03746.x>
- Hillen, J. (2019). Web scraping for food price research. *British Food Journal*, 121(12), 3350–3361. <https://doi.org/10.1108/BFJ-02-2019-0081>
- Jannach, D. (Ed.). (2011). *Recommender systems: An introduction*. Cambridge University Press.
- Kadowaki, T., Yamakata, Y., Mori, S., & Tanaka, K. (2014). Recipe search for blog-type recipe articles based on a user's situation. *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, 497–506.
<https://doi.org/10.1145/2638728.2641327>
- Lawson, R. (2015). *Web scraping with Python: Scrape data from any website with the power of Python*. Packt Publishing.
- Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. Cambridge University Press.
- Nilesh, N., Kumari, M., Hazarika, P., & Raman, V. (2019). Recommendation of Indian Cuisine Recipes Based on Ingredients. *2019 IEEE 35th International Conference on Data Engineering Workshops (ICDEW)*, 96–99.
<https://doi.org/10.1109/ICDEW.2019.00-28>
- Shirai, S. S., Seneviratne, O., Gordon, M. E., Chen, C.-H., & McGuinness, D. L. (2021). Identifying Ingredient Substitutions Using a Knowledge Graph of Food. *Frontiers in Artificial Intelligence*, 3, 621766.
<https://doi.org/10.3389/frai.2020.621766>
- Sukestiyarno, Y. L., Sapolo, H. A., & Sofyan, H. (2023). *Application of Recommendation System on E-Learning Platform Using Content-Based Filtering with Jaccard Similarity and Cosine Similarity Algorithms* [Preprint]. Computer Science and Mathematics.
<https://doi.org/10.20944/preprints202306.1672.v1>
- Tobey, L., Mouzong, C., Angulo, J., Bowman, S., & Manore, M. (2019). How Low-Income Mothers Select and Adapt Recipes and Implications for Promoting Healthy Recipes Online. *Nutrients*, 11(2), 339.
<https://doi.org/10.3390/nu11020339>
- Zhao, B. (2017). Web Scraping. In L. A. Schintler & C. L. McNeely (Eds.), *Encyclopedia of Big Data* (pp. 1–3). Springer International Publishing. https://doi.org/10.1007/978-3-319-32001-4_483-1
- Zhou, B., Yamanaka-Okumura, H., Seki, S., Tatano, H., Adachi, C., & Takeda, E. (2014). What influences appetite more: Eating approaches or cooking methods? *The Journal of Medical Investigation*, 61(1.2), 118–125.
<https://doi.org/10.2152/jmi.61.118>